UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/552,292	04/19/2000	Arch D. Robison	42911329	2880

7590 11/14/2003

JOHN PATRICK WARD
BLAKELY, SOKOLOFF, TAYLOR, & ZAFMAN LLP.
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025

EXAMINER	
GROSS, KENNETH A	

ART UNIT	PAPER NUMBER
2122	

DATE MAILED: 11/14/2003

18

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/552,292	ROBISON, ARCH D.	
	Examiner	Art Unit	
	Kenneth A Gross	2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-6 and 10-17* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-6 and 10-17* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. §§ 119 and 120

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

13)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

a) ☐ The translation of the foreign language provisional application has been received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449) Paper No(s) *4* .
4) ☐ Interview Summary (PTO-413) Paper No(s). _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: .

DETAILED ACTION

1. In view of the Amendment received on September 17th, 2003, the examiner withdraws the final rejection mailed on July 13th, 2003.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1, 2, 10, 13, and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over "How Debuggers Work", Jonathan B. Rosenberg, 1996 (hereinafter Rosenberg) in view of Lo et al. (U.S. Patent Number 6,151,706).

In regard to Claim 1, Rosenberg teaches the well-known concept of the 'program stack', which keeps track of addresses and local variables. Rosenberg teaches: (A) as the program is executing, the state of the stack is analyzed by 'unwinding' the stack at breakpoints to find the current state of the stack (page 136, lines 26-32); (B) partitioning the stack at each point into records or 'frames' that can be set separately (page 136, lines 25-26); (C) Rosenberg teaches storing addresses and variables on the stack, which is done with a push or similar command for storing information onto a stack (page 137, lines 28-32). These 'push' commands sets a component, or frame, of the stack by pushing information (such as addresses and variables) onto the stack, updating the state of the stack. Rosenberg does not teach eliminating partial redundancy by placing the operations stated in step (C) into the code. Lo, however, does teach

removing partial redundancy by rearranging code (Column 3, lines 23-35). Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to analyze the state of a data structure at different program points, where the structure state is broken up into components and set, as taught by Rosenberg, and then placing the set instructions in the code so as to eliminate partial redundancy, as taught by Lo, since this would allow for a more optimized program overall.

In regard to Claim 2, the examiner takes official notice that the stack data structure, as described by Rosenberg above, stores items on a first-in last-out basis, since this is the nature of a stack.

In regard to Claim 10, Rosenberg teaches the well-known concept of the 'program stack', which keeps track of addresses and local variables. Rosenberg teaches: (A) as the program is executing, the state of the stack is analyzed by 'unwinding' the stack at breakpoints to find the current state of the stack (page 136, lines 26-32); (B) partitioning the stack at each point into records or 'frames' that can be set separately (page 136, lines 25-26); (C) Rosenberg teaches storing addresses and variables on the stack, which is done with a push or similar command for storing information onto a stack (page 137, lines 28-32). These 'push' commands sets a component, or frame, of the stack by pushing information (such as addresses and variables) onto the stack, updating the state of the stack. Rosenberg does not teach computing placement of said operations to eliminate partial redundancy and inserting the set of operations and computed placements. Lo, however, does teach computing placement of code through a series of code motions in order to eliminate partial redundancy (Column 2, lines 10-29). Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to analyze the

state of a data structure at different program points, where the structure state is broken up into components and set, as taught by Rosenberg, and then computing placement of the set instructions in the code so as to eliminate partial redundancy and placing the set instructions in the code according to the computed placement, as taught by Lo, since this would allow for a more optimized program overall.

Claim 13 is a medium Claim that corresponds with Claim 10 and is rejected for the same reasons as Claim 10, where Lo teaches a medium to carry out said method (Figure 11).

In regard to Claim 14, the limitations of Claim 14 have already been addressed in Claim 2 and Claim 14 is rejected for the same reasons as Claim 2.

3. Claims 3 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over "How Debuggers Work", Jonathan B. Rosenberg, 1996 (hereinafter Rosenberg) in view of Lo et al. (U.S. Patent Number 6,151,706) and further in view of Gordon et al. (U.S. Patent Number 6,507,805).

In regard to Claim 3, Rosenberg and Lo teach the device of Claim 2, but do not teach representing the stack as a tree of nodes. Gordon however, teaches a call stack tree (Column 18, lines 25-27), where each path traverses the tree towards the root (Fig 14, item 1358) and each node represents a component of the state (Column 18, lines 25-32). Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention to analyze the state of a data structure at different program points, where the structure state is broken up into components and set, as taught by Rosenberg, and then placing the set instructions in the code so as to eliminate partial redundancy, as taught by Lo, and then representing the state of the data

structure as a tree, as taught by Gordon, since this would allow for a more organized method of storing states, and a more efficient method for searching states.

In regard to Claim 15, the limitations of Claim 15 have already been addressed in Claim 3 and Claim 15 is rejected for the same reasons as Claim 3.

4. Claims 4, 5, 11, 12, 16, and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over "How Debuggers Work", Jonathan B. Rosenberg, 1996 (hereinafter Rosenberg) in view of Lo et al. (U.S. Patent Number 6,151,706) and further in view of Dunn et al. (U.S. Patent Number 6,247,172).

In regard to Claim 4, Rosenberg and Lo teach the device of Claim 2, but do not teach that the data structure represents actions to be taken if an exception occurs. Dunn, however, teaches a stack that contains actions to be taken when an exception occurs (Column 1, lines 66-67 and Column 2, lines 1-8). Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention to analyze the state of a data structure at different program points, where the structure state is broken up into components and set, as taught by Rosenberg and the structure represents actions to be taken if an exceptional situation arises as taught by Dunn, and then placing the set instructions in the code so as to eliminate partial redundancy, as taught by Lo, since this would allow for more efficient error handling in programs.

In regard to Claim 5, Rosenberg teaches storing return addresses and local variables before functions and routines (page 136, 9-13). Since any function or routine is capable of causing an error, any program point selected will be immediately before an instruction that might cause an exceptional situation.

In regard to Claim 11, Dunn teaches a stack structure that is used for exception handling, which stores the state of a program at a certain point (Column 1, lines 66-67 and Column 2, lines 1-8).

In regard to Claim 12, Dunn teaches an exception handling data structure referred to as a 'context structure' that stores the program state at a certain point, and is an element on the exception-handling stack. It is obvious that a pointer to the exception-handling stack would be a component, since a pointer would be necessary to access the stack in the case of an exception being thrown.

In regard to Claim 16, the limitations of Claim 16 have already been addressed in Claim 4 and Claim 16 is rejected for the same reasons as Claim 4.

In regard to Claim 17, the limitations of Claim 17 have already been addressed in Claim 5 and Claim 17 is rejected for the same reasons as Claim 5.

5. Claims 6 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over "How Debuggers Work", Jonathan B. Rosenberg, 1996 (hereinafter Rosenberg) in view of Lo et al. (U.S. Patent Number 6,151,706) and further in view of Dunn et al. (U.S. Patent Number 6,247,172) and Gordon et al. (U.S. Patent Number 6,507,805).

In regard to Claim 6, Dunn teaches restoring the state of the program before an exception occurs (Column 1, lines 66-67 and Column 2, lines 1-8), and Rosenberg teaches 'unwinding' the stack at breakpoints to find the current state of the stack (page 136, lines 26-32). Thus, if the state of the stack after restoring the stack to its pre-exception state is converted into a tree (which is a graph) structure as taught by Gordon, the exceptional path would not exist, since it no longer exists in the stack.

In regard to Claim 18, the limitations of Claim 18 have already been addressed in Claim 6 and Claim 18 is rejected for the same reasons as Claim 6.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kenneth A Gross whose telephone number is (703) 305-0542. The examiner can normally be reached on Mon-Fri 7:30-5.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q Dam can be reached on (703) 305-4552. The fax phone number for the organization where this application or proceeding is assigned is (703) 746-7239.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

KAG

TUAN DAM
SUPERVISORY PATENT EXAMINER